WIKIPEDIA

# Cross entropy

In information theory, the **cross entropy** between two probability distributions $p$ and $q$ over the same underlying set of events measures the average number of bits needed to identify an event drawn from the set, if a coding scheme is used that is optimized for an "unnatural" probability distribution $q$, rather than the "true" distribution $p$.

The cross entropy for the distributions $p$ and $q$ over a given set is defined as follows:

$$H(p,q) = \mathrm{E}_p[-\log q] = H(p) + D_{\mathrm{KL}}(p\|q),$$

where $H(p)$ is the entropy of $p$, and $D_{\mathrm{KL}}(p\|q)$ is the Kullback–Leibler divergence of $q$ from $p$ (also known as the *relative entropy* of *p* with respect to *q* — note the reversal of emphasis).

For discrete $p$ and $q$ this means

$$H(p,q) = -\sum_x p(x)\,\log q(x).$$

The situation for continuous distributions is analogous. We have to assume that $p$ and $q$ are absolutely continuous with respect to some reference measure $r$ (usually $r$ is a Lebesgue measure on a Borel σ-algebra). Let $P$ and $Q$ be probability density functions of $p$ and $q$ with respect to $r$. Then

$$-\int_X P(x)\,\log Q(x)\,dr(x) = \mathrm{E}_p[-\log Q].$$

NB: The notation $H(p,q)$ is also used for a different concept, the joint entropy of $p$ and $q$.

## Contents



## Motivation

In information theory, the Kraft–McMillan theorem establishes that any directly decodable coding scheme for coding a message to identify one value $x_i$ out of a set of possibilities $X$ can be seen as representing an implicit probability distribution $q(x_i) = 2^{-l_i}$ over $X$, where $l_i$ is the length of the code for $x_i$ in bits. Therefore, cross entropy can be interpreted as the expected message-length per datum when a wrong distribution $Q$ is assumed while the data actually follows a distribution $P$. That is why the expectation is taken over the probability distribution $P$ and not $Q$.

$$H(p,q) = \mathrm{E}_p[l_i] = \mathrm{E}_p\left[\log\frac{1}{q(x_i)}\right]$$

$$H(p,q) = \sum_{x_i} p(x_i)\,\log\frac{1}{q(x_i)}$$

$$H(p,q) = -\sum_x p(x)\,\log q(x).$$

# Estimation

There are many situations where cross-entropy needs to be measured but the distribution of $p$ is unknown. An example is language modeling, where a model is created based on a training set $T$, and then its cross-entropy is measured on a test set to assess how accurate the model is in predicting the test data. In this example, $p$ is the true distribution of words in any corpus, and $q$ is the distribution of words as predicted by the model. Since the true distribution is unknown, cross-entropy cannot be directly calculated. In these cases, an estimate of cross-entropy is calculated using the following formula:

$$H(T,q) = -\sum_{i=1}^{N}\frac{1}{N}\log_2 q(x_i)$$

where $N$ is the size of the test set, and $q(x)$ is the probability of event $x$ estimated from the training set. The sum is calculated over $N$. This is a Monte Carlo estimate of the true cross entropy, where the training set is treated as samples from $p(x)$.

# Relation to log-likelihood

In classification problems we want to estimate the probability of different outcomes. If the estimated probability of outcome $i$ is $q_i$, while the frequency (empirical probability) of outcome $i$ in the training set is $p_i$, and there are N samples in the training set, then the likelihood of the training set is

$$\prod_i q_i^{Np_i}$$

so the log-likelihood, divided by $N$ is

$$\frac{1}{N}\log\prod_i q_i^{Np_i} = \sum_i p_i \log q_i = -H(p,q)$$

so that maximizing the likelihood is the same as minimizing the cross entropy

# Cross-entropy minimization

Cross-entropy minimization is frequently used in optimization and rare-event probability estimation; see the cross-entropy method

When comparing a distribution $q$ against a fixed reference distribution $p$, cross entropy and KL divergence are identical up to an additive constant (since $p$ is fixed): both take on their minimal values when $p = q$, which is $0$ for KL divergence, and $\mathrm{H}(p)$ for cross entropy.[1] In the engineering literature, the principle of minimising KL Divergence (Kullback's "Principle of Minimum Discrimination Information") is often called the **Principle of Minimum Cross-Entropy** (MCE), or **Minxent**.

However, as discussed in the article *Kullback–Leibler divergence*, sometimes the distribution $q$ is the fixed prior reference distribution, and the distribution $p$ is optimised to be as close to $q$ as possible, subject to some constraint. In this case the two minimisations are *not* equivalent. This has led to some ambiguity in the literature, with some authors attempting to resolve the inconsistency by redefining cross-entropy to be $D_{\mathrm{KL}}(p\|q)$, rather than $H(p,q)$.

# Cross-entropy error function and logistic regression

Cross entropy can be used to define a loss function in machine learning and optimization. The true probability $p_i$ is the true label, and the given distribution $q_i$ is the predicted value of the current model.

More specifically, let us consider logistic regression, which (in its most basic form) deals with classifying a given set of data points into two possible classes generically labelled $0$ and $1$. The logistic regression model thus predicts an output $y \in \{0, 1\}$, given an input vector $\mathbf{x}$. The probability is modeled using the logistic function $g(z) = 1/(1+e^{-z})$. Namely, the probability of finding the output $y = 1$ is given by

$$q_{y=1} \ = \ \hat{y} \ \equiv \ g(\mathbf{w} \cdot \mathbf{x}) \ = 1/(1+e^{-\mathbf{w} \cdot \mathbf{x}}),$$

where the vector of weights $\mathbf{w}$ is optimized through some appropriate algorithm such as gradient descent. Similarly, the complementary probability of finding the output $y = 0$ is simply given by

$$q_{y=0} \ = \ 1 - \hat{y}$$

The true (observed) probabilities can be expressed similarly as $p_{y=1} = y$ and $p_{y=0} = 1 - y$.

Having set up our notation, $p \in \{y, 1-y\}$ and $q \in \{\hat{y}, 1-\hat{y}\}$, we can use cross entropy to get a measure of dissimilarity between $p$ and $q$:

$$H(p,q) \ = \ -\sum_i p_i \log q_i \ = \ -y \log \hat{y} - (1-y) \log(1-\hat{y})$$

The typical cost function that one uses in logistic regression is computed by taking the average of all cross-entropies in the sample. For example, suppose we have $N$ samples with each sample indexed by $n = 1, \dots, N$. The loss function is then given by:

$$J(\mathbf{w}) \ = \ \frac{1}{N} \sum_{n=1}^{N} H(p_n, q_n) \ = \ -\frac{1}{N} \sum_{n=1}^{N} \left[ y_n \log \hat{y}_n + (1-y_n) \log(1-\hat{y}_n) \right],$$

where $\hat{y}_n \equiv g(\mathbf{w} \cdot \mathbf{x}_n) = 1/(1+e^{-\mathbf{w} \cdot \mathbf{x}_n})$, with $g(z)$ the logistic function as before.

The logistic loss is sometimes called cross-entropy loss. It is also known as log loss (In this case, the binary label is often denoted by {-1,+1}).[2]

# See also

- Cross-entropy method
- Logistic regression
- Conditional entropy
- Maximum likelihood estimation

# References


1. Ian Goodfellow, Yoshua Bengio, and Aaron Courville (2016). Deep Learning. MIT Press. Online (http://www.deeplearningbook.org)
2. Murphy, Kevin (2012). *Machine Learning: A Probabilistic Perspective*. MIT. ISBN 978-0262018029.

- de Boer, Pieter-Tjerk; Kroese, Dirk P.; Mannor, Shie; Rubinstein, Reuven Y. (February 2005). "A Tutorial on the Cross-Entropy Method" (PDF). *Annals of Operations Research* (pdf). **134** (1). pp. 19–67. doi:10.1007/s10479-005-5724-z. ISSN 1572-9338.

# External links

- What is cross-entropy, and why use it?
- Cross Entropy